

PRACTICAL INTRO TO GOLANG

Tomáš Karásek

Coding Sauna 28.3.2022 @TribeTampere

DESIGN GOALS

- Go combines the development speed of working in a dynamic language like Python with the performance and safety of a compiled language like C or C++
- Developed in Google to address the needs of Internet services (fast change, dependencies, concurrency)
- Simple to learn and use (familiar syntax, garbage collection, static linking, light OOP, easy tooling)

TECHNICAL CHARACTERISTICS

- (open /hello in vscode)
- Compiled (Code -> Executable binary for target platform)
- Golang code is linked statically
- Compiled binaries have little dependencies
- Pointers, garbage-collected
- Very low-level, but has concurrency and map in core (unlike C,C++ and Java)

POPULARITY

- Well paid (2020, 2021), loved (2020, 2021), but still not there
- Supported in Github Copilot, in Google Cloud Functions

TOOLING

- most come as subcommands in the same binary as the compiler
- go build - compiler
- go mod - dependency management
- go fmt - formatter (show in /hello/)
- go test - testing (show in /hello/)
- go doc - generates HTML documentation

CODING EXAMPLE `/tickers/`

- show `BTC-USD order book on Coinbase`
- show API get of BTC-USD from Coinbase
- getting JSON from API, struct, tags, `json.Unmarshal`
- `CoinbaseGetter BTC-USD`
- print prices from more exchange
- print prices from more exchange concurrently
- compute average concurrently
- compare times for sync and async

WRITTEN IN GO

Usually sysadmin, "DevOps" tools

- Kubernetes, Docker
- Prometheus, Influxdb
- Terraform, any software from HashiCorp

VS JAVA, C++, PYTHON

- no classes, no inheritance, but struct can be composed, and can have "methods"
- Java: no VM, map, concurrency out of the box
- C++: garbage col, static linking, no generics until recently
- Python: static typing, compiled, concurrency, faster